Amendment to the Claims

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1.   (Currently amended) A method ~~of determining how a region of a data structure in an application evolves,~~ comprising steps of:

deriving a suspect region exhibiting problematic data structure evolution in a running application;

periodically traversing only selected ~~constant-sized~~ subgraphs of ~~a full graph in~~ the suspect region ~~in the application~~ to a depth of one, in order to detect data structure changes ~~of patterns in the subgraphs while the application is running~~, wherein a data structure is a subgraph of an object reference graph snapshot and the subgraph comprises nodes that own constituents;

characterizing the data structure changes;

updating a histogram of the suspect region with the characterizations of the data structure changes;

using ~~these~~ the characterized data structure changes to describe, characterize, and identify ~~changes to~~ an evolutionary trend of the suspect region as a whole; and

reporting the characterized changes to the region to an analysis agent.

2.   (Cancelled)

3.   (Original) The method of claim 1 used to detect one of the following changes to a region: additions to a region; removals from a region; and internal restructuring within a region.

4.   (Currently amended) The method of claim 1 wherein the selected subgraphs to traverse are derived by

computing ~~the~~ region ~~key~~ keys for the constituents of the data structure; and

identifying ~~the~~ a unique set of paths from owner proxy to change proxy as the set of traversals.

5.   (Currently amended) The method of claim 4 wherein the traversals are shortened by

identifying a subpath of ~~the~~ a path which is unlikely to change as the region evolves; and

trimming the path to exclude ~~the~~ parts of the path which are unlikely to change.

6.   (Currently amended) The method of claim 1 wherein determining how [[a]] the suspect region ~~of a data structure in an application~~ evolves is a continuous and adaptive process.

7.   (Currently amended) The method of claim 6 wherein the process is made continuous and adaptive by

identifying a set of desired updates; and

adjusting ~~the period~~ an interval in between traversals based on whether the desired updates have been witnessed.

8.   (Currently amended) The method of claim 6 wherein the process is made continuous and adaptive by

identifying a set of desired updates; and

adjusting ~~the~~ a_frequency of sampling any one traversal based on whether that traversal has detected desired updates.

9.   (Original) The method of claim 6 wherein the process is made continuous and adaptive by implementing one of the following procedures based on the result of performing a traversal: adding new traversals; removing existing traversals; and modifying the path of existing traversals.

10.   (Currently amended) The method of claim 1 further comprising

updating qualitative characterizations of the suspect regions under analysis based on structural changes to the regions as a whole.

11.   (Original) The method of claim 1 further comprising

updating quantitative characterizations of the regions under analysis based on structural changes to the regions as a whole.

12-13 (cancelled)

14. (New) The method of claim 6 further comprising using the evolutionary trend to update rankings of regions exhibiting problematic data structure evolution.

15. (New) The method of claim 15 wherein updating the histogram comprises:

allocating one row for each suspect region; and

storing, for each suspect region:

a ranking of the suspect region;

a proxy-size of the suspect region; and

a summary of attributes of a last change detected and previous n changes detected.

16. (New) The method of claim 15 wherein storing the summary of attributes comprises:

classifying the suspect region as a monotonic grower if only addition updates were detected;

classifying the suspect region as an oscillator if a substantially equal distribution of addition and removal updates were detected;

classifying the region as a shrinker if only removal updates were detected; and

classifying the region as flatliner if no updates were detected.